


```
LL      IIIIII  BBBB BBBB  FFFFFFFF  AAAAAA  000000  LL
LL      IIIIII  BBBB BBBB  FFFFFFFF  AAAAAA  000000  LL
LL      II      BB      BB  FF      AA      AA  00      00  LL
LL      II      BB      BB  FF      AA      AA  00      00  LL
LL      II      BB      BB  FF      AA      AA  00      00  LL
LL      II      BBBB BBBB  FFFFFFFF  AA      AA  00      00  LL
LL      II      BBBB BBBB  FFFFFFFF  AA      AA  00      00  LL
LL      II      BB      BB  FF      AAAAAAAAAA  00      00  LL
LL      II      BB      BB  FF      AAAAAAAAAA  00      00  LL
LL      II      BB      BB  FF      AA      AA  00      00  LL
LL      II      BB      BB  FF      AA      AA  00      00  LL
LL      IIIIII  BBBB BBBB  FF      AA      AA  000000  LL
LL      IIIIII  BBBB BBBB  FF      AA      AA  000000  LL
LLLLLLLLLLLL  IIIIII  BBBB BBBB  FF      AA      AA  000000  LLLLLLLLLL
LLLLLLLLLLLL  IIIIII  BBBB BBBB  FF      AA      AA  000000  LLLLLLLLLL
                                                    ....
                                                    ....
                                                    ....
                                                    ....
```

```
LL      IIIIII  SSSSSSSS
LL      IIIIII  SSSSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SSSSSS
LL      II      SSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      IIIIII  SSSSSSSS
LL      IIIIII  SSSSSSSS
LLLLLLLLLLLL  SSSSSSSS
LLLLLLLLLLLL  SSSSSSSS
```

```
1 0001 0 MODULE LIBSYS_FAOL (
2 0002 0
3 0003 0 IDENT = '1-009' ! File: LIBFAOL.B32 EDIT RKR1009
4 0004 0
5 0005 0 ) =
6 0006 1 BEGIN
7 0007 1
8 0008 1 *****
9 0009 1 *
10 0010 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
11 0011 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
12 0012 1 * ALL RIGHTS RESERVED.
13 0013 1 *
14 0014 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
15 0015 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
16 0016 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
17 0017 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
18 0018 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
19 0019 1 * TRANSFERRED.
20 0020 1 *
21 0021 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
22 0022 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
23 0023 1 * CORPORATION.
24 0024 1 *
25 0025 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
26 0026 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
27 0027 1 *
28 0028 1 *****
29 0029 1
30 0030 1
31 0031 1
32 0032 1 ++
33 0033 1 FACILITY: General Library Support
34 0034 1
35 0035 1 ABSTRACT:
36 0036 1
37 0037 1 This routine calls $FAOL for the caller, returning the
38 0038 1 resulting string using the semantics of the caller's string.
39 0039 1 The length of the resultant string is limited to 256 bytes max and
40 0040 1 truncation may occur.
41 0041 1
42 0042 1
43 0043 1 ENVIRONMENT: VAX-11 User Mode
44 0044 1
45 0045 1 AUTHOR: R. Reichert, CREATION DATE: 20-SEPT-1979
46 0046 1
47 0047 1 MODIFIED BY:
48 0048 1
49 0049 1 1-001 - Original. RKR 20-SEPT-1979
50 0050 1 1-002 - Change name to LIBSYS_FAOL. RKR 15-OCT-1979
51 0051 1 1-003 - Change logic for status returned. Improve comments.
52 0052 1 RKR 20-OCT-79
53 0053 1 1-004 - Correct case where P1 is an actual parameter of value 0.
54 0054 1 RKR 29-OCT-79
55 0055 1 1-005 - Improve description of routine. RKR 1-NOV-79
56 0056 1 1-006 - Change STR$ codes to LIB$ codes. RKR 22-JAN-80
57 0057 1 1-007 - Enhance to recognize additional classes of string descriptors
```



```
58      0058 1 | on output by always getting data into internal buffer and
59      0059 1 | using LIB$SCOPY_DXDX6 to deliver to caller's buffer.
60      0060 1 | LIB$ANALYZE_SDESC_R3 is used to compute the number of bytes
61      0061 1 | actually copied after the copy operation completes.
62      0062 1 | Change to use LIB$SCOPY_DXDX6 instead of STR$COPY_DX. This
63      0063 1 | avoids having to change STR$ statuses to LIB$ statuses and
64      0064 1 | eliminates need for a handler.
65      0065 1 | RKR 29-MAY-1981.
66      0066 1 | 1-008 - Add special-case code to process string descriptors that
67      0067 1 | "read" like fixed string descriptors. RKR 7-OCT-1981
68      0068 1 | 1-009 - Redirect jsb's from LIB$ANALYZE_SDESC_R3 to
69      0069 1 | LIB$ANALYZE_SDESC_R2. Do copying using LIB$SCOPY_R_DX6.
70      0070 1 | RKR 18-NOV-1981
71      0071 1 | --
72      0072 1 |
73      0073 1 | <BLF/PAGE>
```

```
.. 75      0074 1 |
.. 76      0075 1 | SWITCHES:
.. 77      0076 1 |
.. 78      0077 1 |
.. 79      0078 1 | SWITCHES ADDRESSING MODE
.. 80      0079 1 |         (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
.. 81      0080 1 |
.. 82      0081 1 |
.. 83      0082 1 | LINKAGES:
.. 84      0083 1 |
.. 85      0084 1 | REQUIRE 'RTLIN:STRLNK';
.. 86      0269 1 |
.. 87      0270 1 | TABLE OF CONTENTS:
.. 88      0271 1 |
.. 89      0272 1 |
.. 90      0273 1 | FORWARD ROUTINE
.. 91      0274 1 |     LIB$SYS_FAOL;           ! Formated ASCII output
.. 92      0275 1 |
.. 93      0276 1 |
.. 94      0277 1 | INCLUDE FILES:
.. 95      0278 1 |
.. 96      0279 1 |
.. 97      0280 1 | REQUIRE 'RTLIN:RTLPSECT';   ! Macros for defining psects
.. 98      0375 1 |
.. 99      0376 1 | LIBRARY 'RTLSTARLE';       ! System symbols and macros
100      0377 1 |
101      0378 1 |
102      0379 1 | MACROS:
103      0380 1 |
104      0381 1 |     NONE
105      0382 1 |
106      0383 1 | EQUATED SYMBOLS:
107      0384 1 |
108      0385 1 | LITERAL
109      0386 1 |     LOCAL_BUF_SIZE = 256 ;
110      0387 1 |
111      0388 1 | PSECTS:
112      0389 1 |
113      0390 1 | DECLARE_PSECTS (LIB);       ! Declare psects for LIB$ facility
114      0391 1 |
115      0392 1 | OWN STORAGE:
116      0393 1 |
117      0394 1 |     NONE
118      0395 1 |
119      0396 1 | EXTERNAL REFERENCES:
120      0397 1 |
121      0398 1 |
122      0399 1 | EXTERNAL ROUTINE
123      0400 1 |     LIB$ANALYZE_SDESC_R2 : LIB$ANALYZE_SDESC_JSB_LINK, ! Extract length
124      0401 1 |                                                              ! and address of
125      0402 1 |                                                              ! 1st data byte
126      0403 1 |                                                              ! from descriptor
.. 127      0404 1 |     LIB$COPY_R_DX6 : STRING_JSB; ! Copy string
```

```
129 0405 1 GLOBAL ROUTINE LIB$SYS_FAOL (          ! Formated ASCII output
130 0406 1
131 0407 1
132 0408 1     CTRSTR,          ! addr of string desc. for ASCII control string
133 0409 1
134 0410 1     OUTLEN,        ! addr of word in which to store output string
135 0411 1          ! length (Optional)
136 0412 1
137 0413 1     OUTBUF,        ! addr of output buffer string descriptor
138 0414 1
139 0415 1     PRMLST         ! addr of list of longword parameters
140 0416 1
141 0417 1          ! For additional description of parameters,
142 0418 1          ! see $FAOL documentation.
143 0419 1
144 0420 1          ) : =
145 0421 1
146 0422 1 ++
147 0423 1 FUNCTIONAL DESCRIPTION:
148 0424 1
149 0425 1     This routine calls $FAOL for the caller, returning the
150 0426 1     resulting string using the semantics of the caller's string.
151 0427 1     The length of the resultant string is limited to 256 bytes max and
152 0428 1     truncation may occur.
153 0429 1
154 0430 1     See $FAOL description
155 0431 1
156 0432 1 FORMAL PARAMETERS:
157 0433 1
158 0434 1     CTRSTR.rt.dx      addr of string desc. for ASCII control string
159 0435 1
160 0436 1     OUTLEN.wm.r         addr of word in which to store output string
161 0437 1          length
162 0438 1          (Optional parameter)
163 0439 1
164 0440 1     OUTBUF.wt.dx       addr of output buffer string descriptor
165 0441 1
166 0442 1     PRMLST.ra.??      addr of a list of longword parameters
167 0443 1
168 0444 1          ! For additional description of parameters,
169 0445 1          ! see $FAOL documentation.
170 0446 1
171 0447 1
172 0448 1
173 0449 1 IMPLICIT INPUTS:
174 0450 1
175 0451 1     NONE
176 0452 1
177 0453 1 IMPLICIT OUTPUTS:
178 0454 1
179 0455 1     NONE
180 0456 1
181 0457 1 COMPLETION CODES:
182 0458 1
183 0459 1     SS$ NORMAL          Procedure successfully completed
184 0460 1     From LIB$COPY R DX6
185 0461 1     LIB$_STRTRO       Success, but source string truncated
```



```
186 0462 1  LIB$-INSVIRMEM  Insufficient virtual memory
187 0463 1  LIB$-INVSTRDES Invalid string descriptor
188 0464 1  LIB$-INVARG   Invalid argument ( usually unsupported class
189 0465 1  of descriptor)
190 0466 1  From $FAOL
191 0467 1  SSS_BUFFEROVF Successfully completed, but formatted output
192 0468 1  string overflowed the output buffer and has
193 0469 1  been truncated.
194 0470 1  SSS_BADPARAM  An invalid directive was specified in the FAO
195 0471 1  control string.
196 0472 1
197 0473 1  SIDE EFFECTS:
198 0474 1
199 0475 1
200 0476 1  --
201 0477 1
202 0478 2  BEGIN
203 0479 2  BUILTIN
204 0480 2  ACTUALCOUNT,
205 0481 2  NULLPARAMETER,
206 0482 2  ACTUALPARAMETER ;
207 0483 2
208 0484 2  LOCAL
209 0485 2  COPY_STATUS,           ! status from copy operation
210 0486 2  STR_STATUS,           ! status from $FAOL call
211 0487 2  LOCAL_BUFF: VECTOR [LOCAL_BUF_SIZE, BYTE], ! local buffer
212 0488 2  BUFFER_DESC : BLOCK [8, BYTE];           ! descr for local buffer
213 0489 2
214 0490 2  MAP
215 0491 2  OUTBUF : REF BLOCK [, BYTE],
216 0492 2  OUTLEN : REF VECTOR [, WORD];
217 0493 2
218 0494 2  !+
219 0495 2  Construct a static descriptor to receive the data from the system
220 0496 2  service call
221 0497 2  --
222 0498 2  BUFFER_DESC [DSC$W_LENGTH] = LOCAL_BUF_SIZE;
223 0499 2  BUFFER_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
224 0500 2  BUFFER_DESC [DSC$B_CLASS] = DSC$K_CLASS_S;
225 0501 2  BUFFER_DESC [DSC$A_POINTER] = LOCAL_BUFF;
226 0502 2
227 0503 2  !+
228 0504 2  Call $FAOL using the appropriate combination of caller's
229 0505 2  parameters, default parameters and local variables
230 0506 2  --
231 0507 2
232 0508 2  STR_STATUS = $FAOL (
233 0509 2  CTRSTR = ACTUALPARAMETER(1),
234 0510 2  OUTLEN = BUFFER_DESC,
235 0511 2  OUTBUF = BUFFER_DESC,
236 0512 2  PRMLST = (IF (ACTUALCOUNT() LSSU 4)
237 0513 2  THEN 0
238 0514 2  ELSE ACTUALPARAMETER(4) )
239 0515 2  ) ;
240 0516 2
241 0517 2  !+
242 0518 2  Copy internal buffer to caller's buffer, letting LIB$SCOPY_R_DX6
```

```
243 0519 2 | worry about its semantics.
244 0520 2 |
245 0521 2 | COPY_STATUS = LIB$COPY_R_DX6 (.BUFFER_DESC [DSC$W_LENGTH],
246 0522 2 | .BUFFER_DESC [DSC$A_POINTER],
247 0523 2 | .OUTBUF);
248 0524 2 |
249 0525 2 |
250 0526 2 | + If caller supplied optional length parameter, update it.
251 0527 2 | To do this accurately, we need to inspect the resulting descriptor
252 0528 2 | as it appears after the copy. Get LIB$ANALYZE_SDESC_R2 to do it.
253 0529 2 | No need to check LIB$ANALYZE_SDESC_R2 status. If OUTBUF descr
254 0530 2 | was bad, LIB$COPY_R_DX6 would have told us about it.
255 0531 2 |
256 0532 2 | IF (NOT NULLPARAMETER(2))
257 0533 2 | THEN
258 0534 2 | BEGIN
259 0535 2 | LOCAL
260 0536 2 | OUTBUF_LEN, ! No. of bytes delivered to caller
261 0537 2 | OUTBUF_ADDR; ! Address of 1st data byte of callers
262 0538 2 | buffer
263 0539 2 | IF .OUTBUF [DSC$B_CLASS] GTRU DSC$K_CLASS_D
264 0540 2 | THEN ! Used generalized length extraction
265 0541 2 | BEGIN
266 0542 2 | LIB$ANALYZE_SDESC_R2 (.OUTBUF; OUTBUF_LEN, OUTBUF_ADDR);
267 0543 2 | END
268 0544 2 |
269 0545 2 | ELSE ! Fetch length directly
270 0546 2 | OUTBUF_LEN = .OUTBUF [DSC$W_LENGTH];
271 0547 2 |
272 0548 2 | OUTLEN [0] = MIN (.BUFFER_DESC [DSC$W_LENGTH], .OUTBUF_LEN);
273 0549 2 |
274 0550 2 | END;
275 0551 2 | +
276 0552 2 | At this point we have 2 statuses that we could return to caller.
277 0553 2 | Determine which one is the "most meaningful".
278 0554 2 |
279 0555 2 | +
280 0556 2 | If original $FAOL call failed, tell him about that
281 0557 2 | else give him the copy status.
282 0558 2 |
283 0559 2 | IF NOT .STR_STATUS THEN RETURN (.STR_STATUS)
284 0560 2 | ELSE RETURN (.COPY_STATUS);
285 0561 2 |
286 0562 2 | END; ! end of LIBSSYS_FAOL
```

```
.TITLE LIBSSYS_FAOL
.IDENT \1-009\
```

```
.EXTRN LIB$ANALYZE_SDESC_R2
.EXTRN LIB$COPY_R_DX6
.EXTRN SYS$FAOL
```

```
.PSECT _LIB$CODE, NOWRT, SHR, PIC, 2
```

```
SE FEFC CE 01FC 0000
9E 0002
```

```
.ENTRY LIBSSYS_FAOL, Save R2,R3,R4,R5,R6,R7,R8
MOVAB -260(SPT), SP
```

```
: 0405
:
```


04	AE	010E0100	8F	DD	00007	PUSHL	#17694976	0498
	04	08	AE	9E	0000D	MOVAB	LOCAL_BUFF, BUFFER_DESC+4	0501
			6C	91	00012	CMPB	(AP), #4	0515
			04	1E	00015	BGEQU	1\$	
			7E	D4	00017	CLRL	-(SP)	
			03	11	00019	BRB	2\$	
		10	AC	DD	0001B	PUSHL	16(AP)	
		04	AE	9F	0001E	PUSHAB	BUFFER_DESC	
		08	AE	9F	00021	PUSHAB	BUFFER_DESC	
		04	AC	DD	00024	PUSHL	4(AP)	
00000000G	00		04	FB	00027	CALLS	#4, SYSS\$FAOL	
	58		50	D0	0002E	MOVL	R0, STR_STATUS	
	57	0C	AC	D0	00031	MOVL	OUTBUF, -R7	0523
	52		57	D0	00035	MOVL	R7, R2	0521
	51	04	AE	D0	00038	MOVL	BUFFER_DESC+4, R1	
	50		6E	3C	0003C	MOVZWL	BUFFER_DESC, R0	
		00000000G	00	16	0003F	JSB	LIB\$COPY_R DX6	
	53		50	D0	00045	MOVL	R0, COPY_STATUS	
	02		6C	91	00048	CMPB	(AP), #2	0532
			28	1F	0004B	BLSSU	6\$	
		08	AC	D5	0004D	TSTL	8(AP)	
			23	13	00050	BEQL	6\$	
	02	03	A7	91	00052	CMPB	3(R7), #2	0539
			0B	1B	00056	BLEQU	3\$	
	50		57	D0	00058	MOVL	R7, R0	0542
		00000000G	00	16	0005B	JSB	LIB\$ANALYZE_SDESC_R2	
			03	11	00061	BRB	4\$	0539
	51		67	3C	00063	MOVZWL	(R7), OUTBUF_LEN	0547
	50		6E	3C	00066	MOVZWL	BUFFER_DESC, R0	0549
	51		50	D1	00069	CMPL	R0, OUTBUF_LEN	
			03	15	0006C	BLEQ	5\$	
	50		51	D0	0006E	MOVL	OUTBUF_LEN, R0	
08	BC		50	B0	00071	MOVW	R0, BOOTLEN	
	04		58	E8	00075	BLBS	STR_STATUS, 7\$	0559
	50		58	D0	00078	MOVL	STR_STATUS, R0	0560
			04	0007B	RET			
	50		53	D0	0007C	MOVL	COPY_STATUS, R0	
			04	0007F	RET			0562

: Routine Size: 128 bytes, Routine Base: _LIB\$CODE + 0000

:	287	0563	1	
:	288	0564	1	END
:	289	0565	1	
:	290	0566	0	ELUDOM

! end of module LIB\$SYS_FAOL

PSECT SUMMARY

Name	Bytes	Attributes
_LIB\$CODE	128	NOVEC, NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC, ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
:_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	10	0	581	00:00.7

COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS\$:LIBFAOL/OBJ=OBJ\$:LIBFAOL MSRC\$:LIBFAOL/UPDATE=(ENHS:LIBFAOL)

: Size: 128 code + 0 data bytes
: Run Time: 00:04.3
: Elapsed Time: 00:18.0
: Lines/CPU Min: 7842
: Lexemes/CPU-Min: 33090
: Memory Used: 72 pages
: Compilation Complete

0206 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

